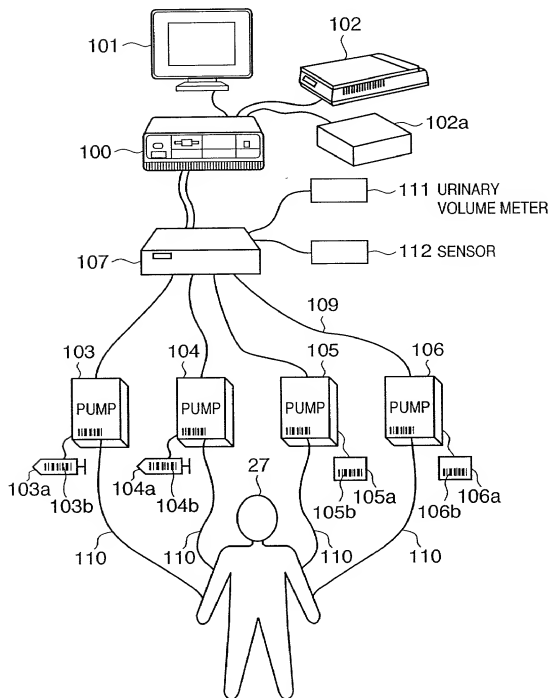


1/25

FIG. 1



0964394 052501

FIG. 2

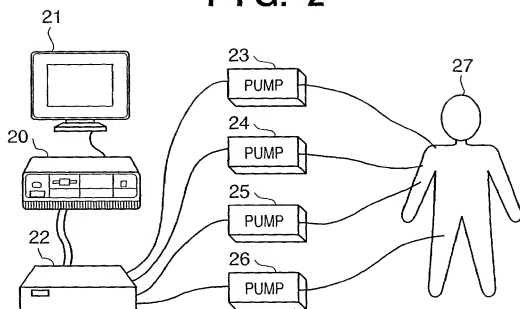


FIG. 3

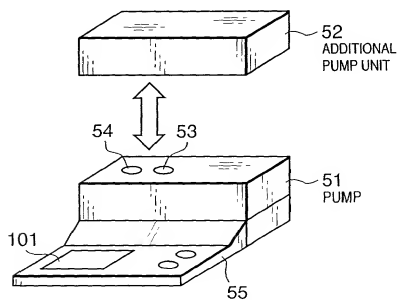


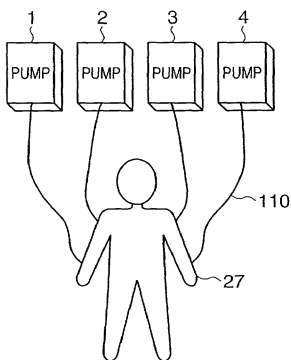
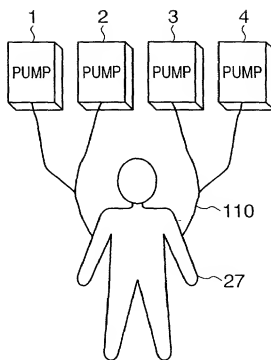
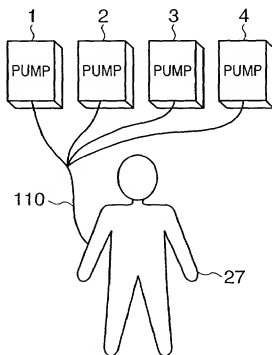
FIG. 4A**FIG. 4B****FIG. 4C**

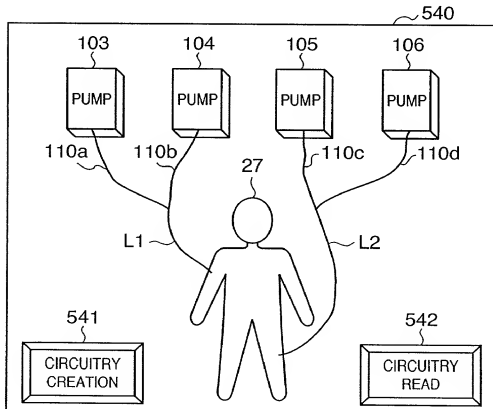
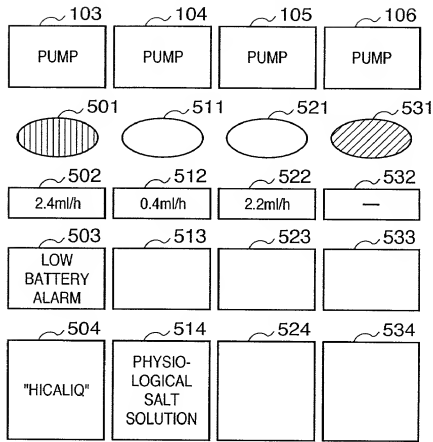
FIG. 5

FIG. 6

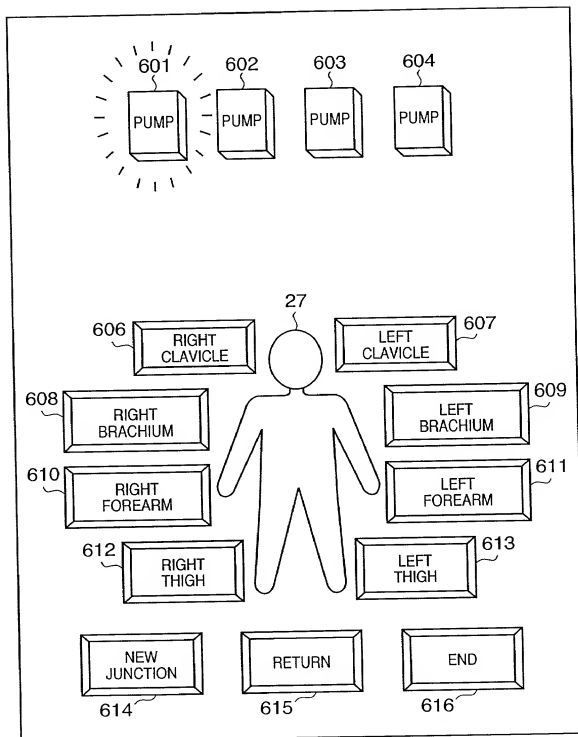


FIG. 7A

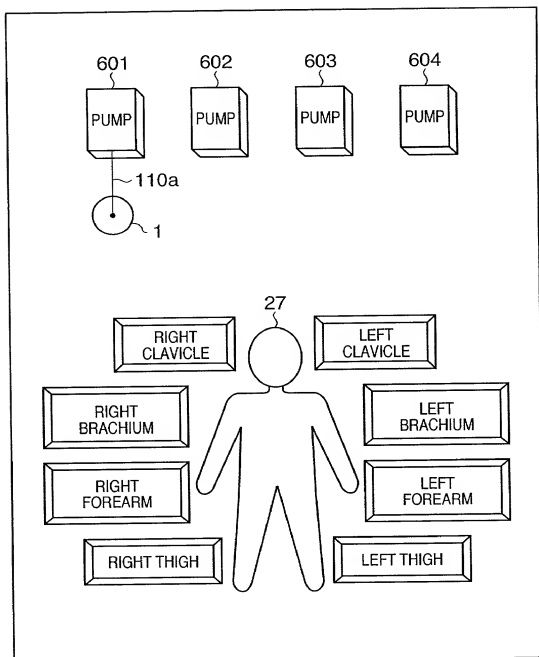


FIG. 7B

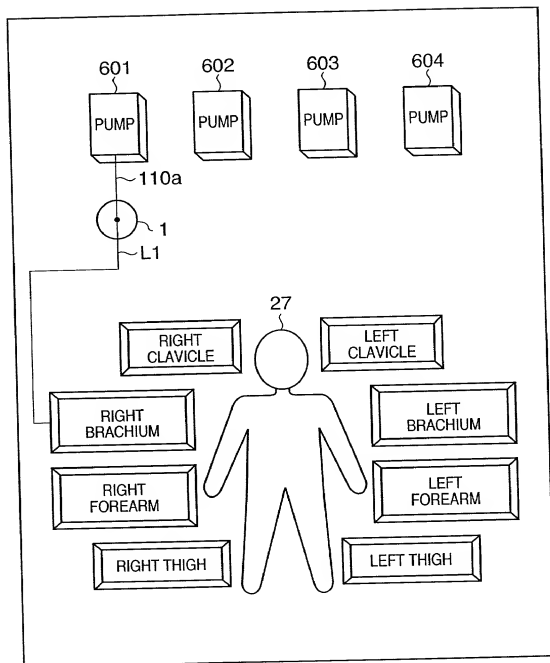


FIG. 7C

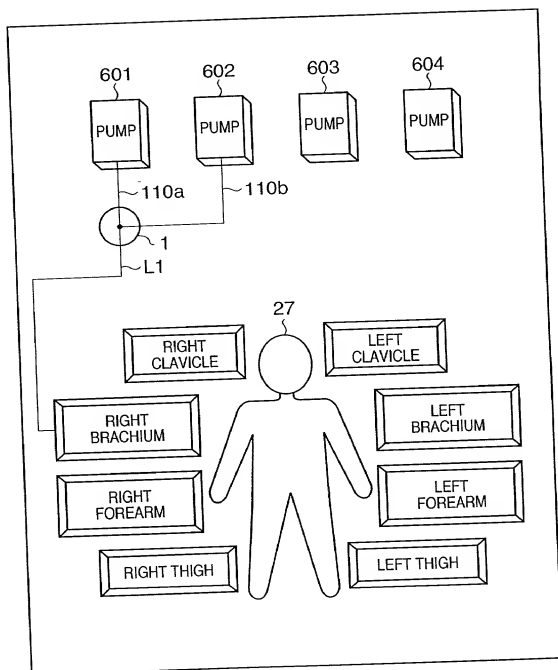


FIG. 7D

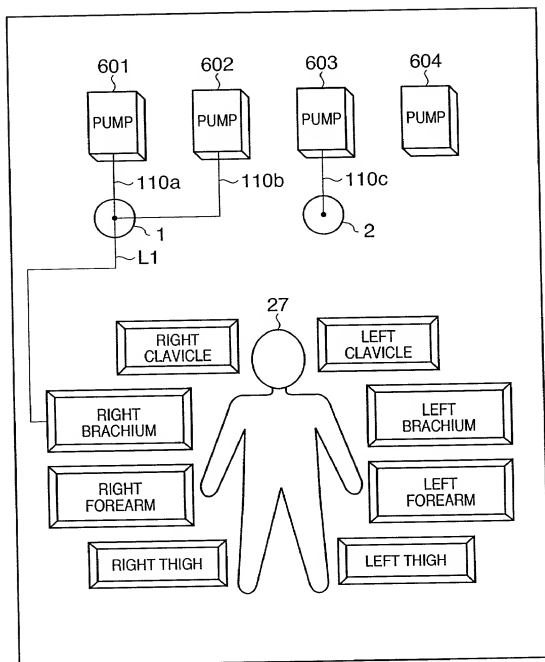


FIG. 7E

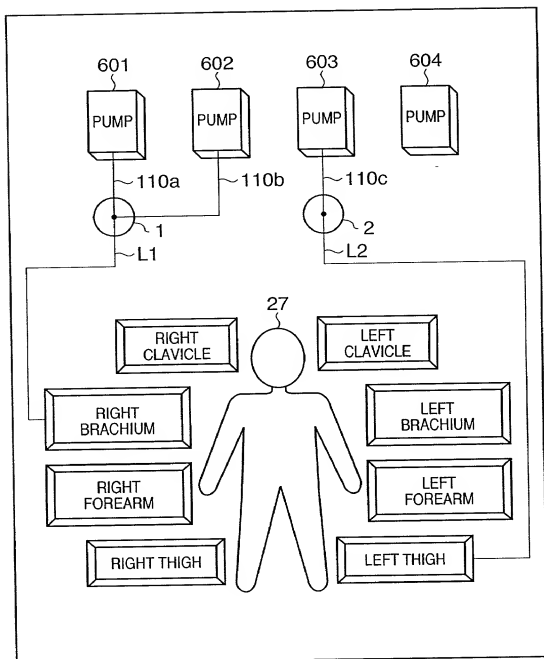


FIG. 7F

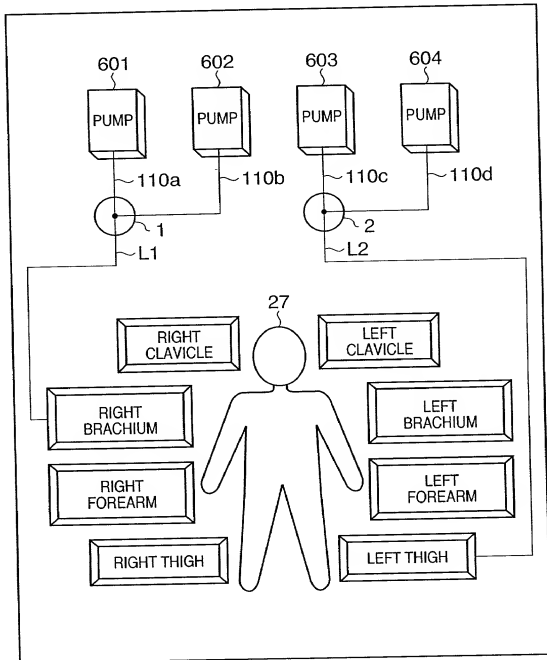


FIG. 7G

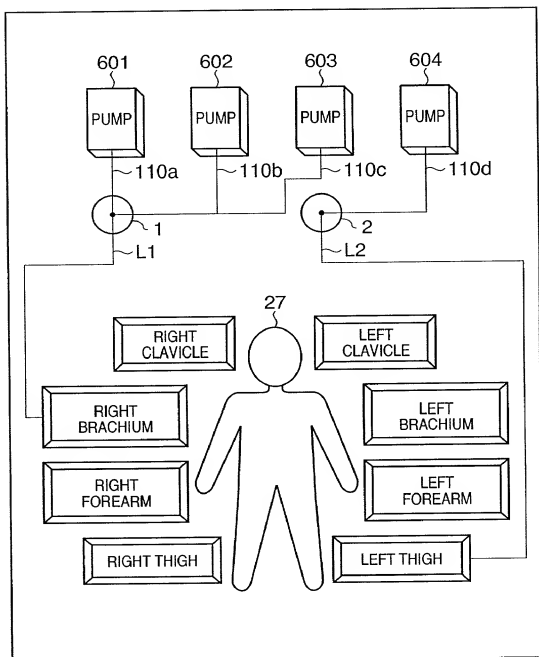


FIG. 8

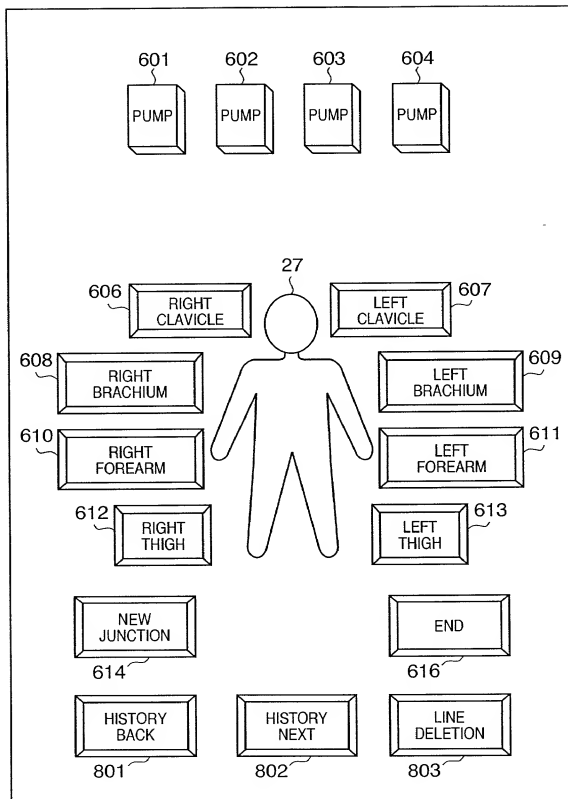


FIG. 9

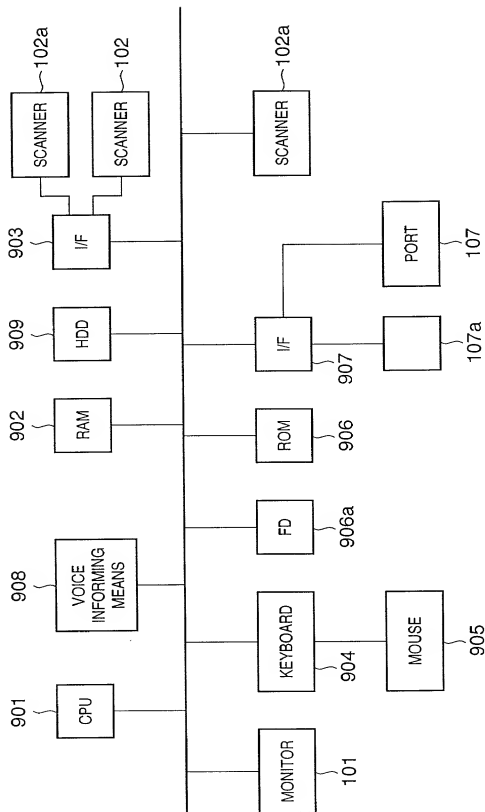


FIG. 10A

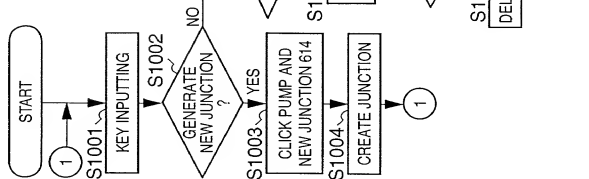


FIG. 10B

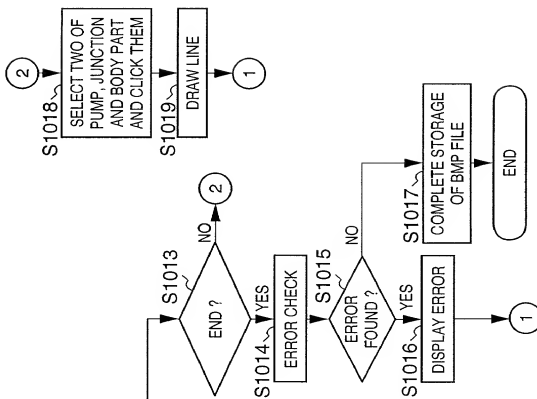


FIG. 11

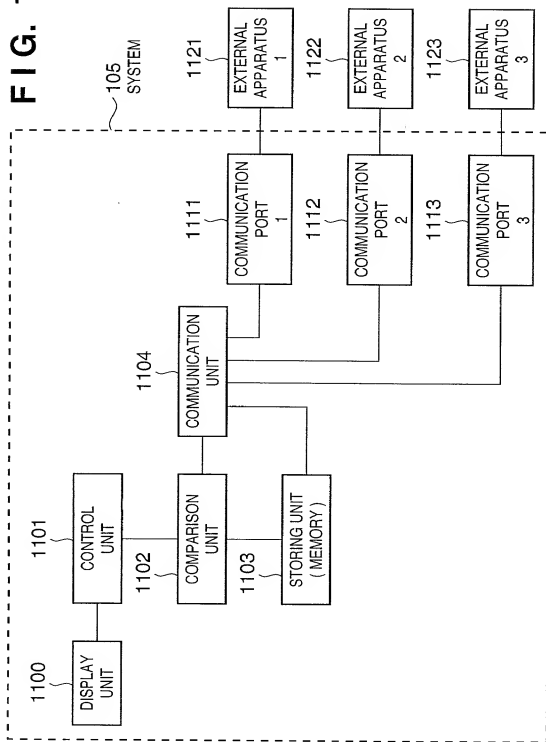


FIG. 12

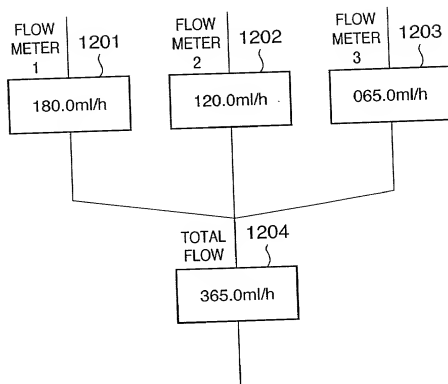


FIG. 14

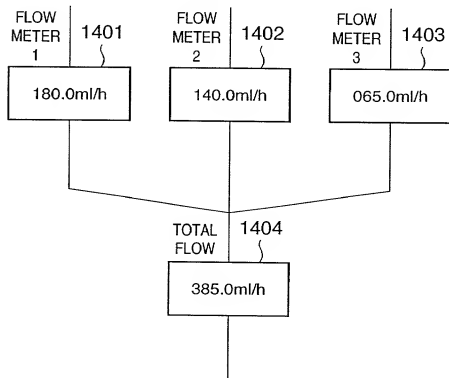
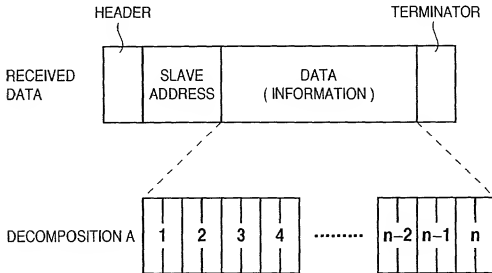


FIG. 15**CALCULATION OF INVERSE BCC**

DECOMPOSITION B = INVERSE (DECOMPOSITION A)

$$BCC1 = 1 \vee 2 \vee 3 \vee 4 \cdot \cdot \cdot \cdot \vee n$$

$$BCC2 = \overline{1} + \overline{2} + \overline{3} + \overline{4} \cdot \cdot \cdot \cdot + \overline{n}$$

LOWER TWO BYTES ARE USED

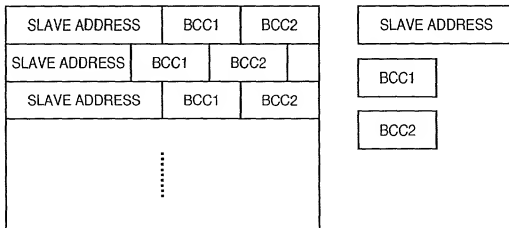
FIG. 16**MEMORY MAP**

FIG. 17**MECHANISM OF HIGH SPEED**

· INVERSE BCC CHECK MODE

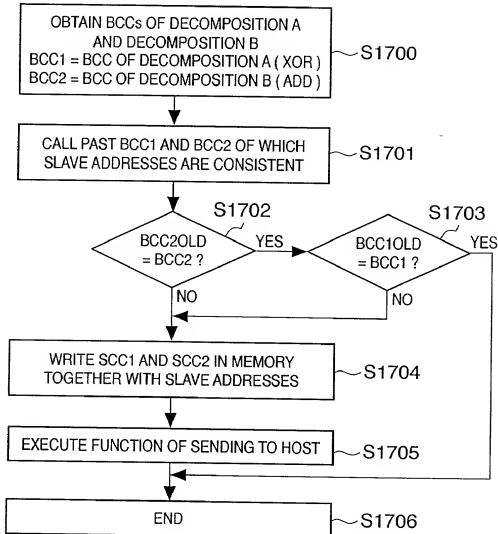
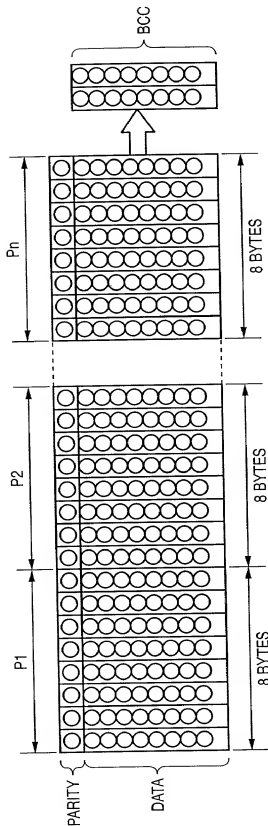


FIG. 18

METHOD OF DETECTING POSITION OF CHANGE



PARITY : 1 BIT FOR CONFIRMING THE NUMBER OF BITS FOR EACH ONE BYTE OF DATA AND
 MAKING AN ADJUSTMENT SO THAT THE TOTAL THEREOF IS ODD OR EVEN NUMBER
 Pn : PARITY PUT TOGETHER FOR EACH EIGHT BYTES OF DATA
 THE POSITION OF CHANGED DATA CAN BE CONFIRMED BY COMPARISON OF Pn

FIG. 19

MEMORY MAP

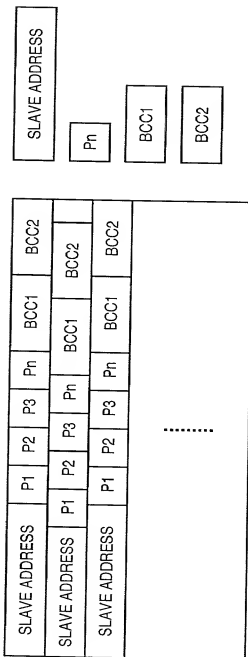


FIG. 20

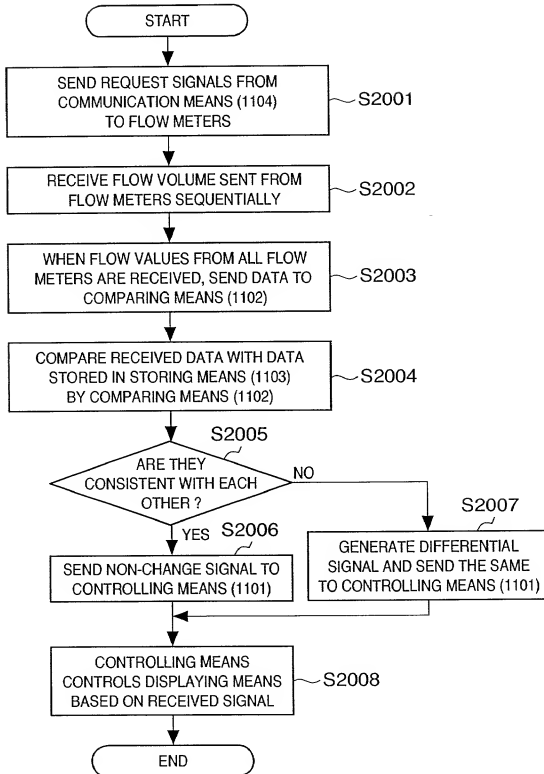


FIG. 21A

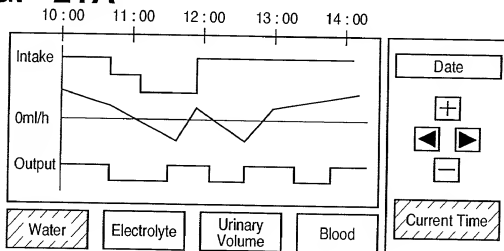


FIG. 21B

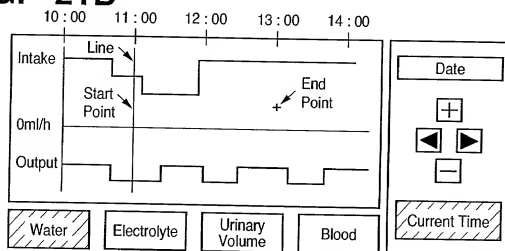


FIG. 21C

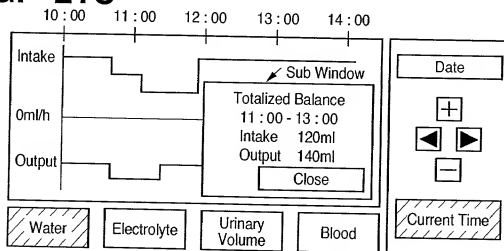


FIG. 22A

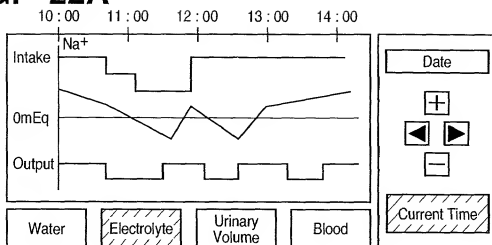


FIG. 22B

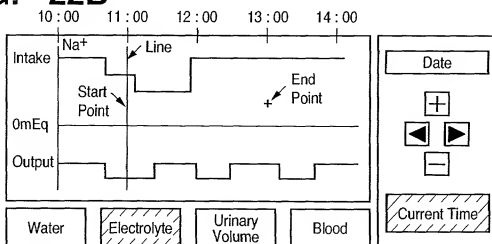


FIG. 22C

